A Graphical Social Topology Model for Multi-Object Tracking

Shan Gao, Xiaogang Chen, Qixiang Ye, *Senior Member, IEEE*, Arjan Kuijper, *Member, IEEE*, Xiangyang Ji, *Member, IEEE*,

***Abstract*—Tracking multiple objects is a challenging task when objects move in groups and occlude each other. Existing methods have investigated the problems of group division and group energy-minimization; however, lacking overall object-group topology modeling limits their ability in handling complex object and group dynamics. Inspired with the social affinity property of moving objects, we propose a Graphical Social Topology (GST) model, which estimates the group dynamics by jointly modeling the group structure and the states of objects using a topological representation. With such topology representation, moving objects are not only assigned to groups, but also dynamically connected with each other, which enables in-group individuals to be correctly associated and the cohesion of each group to be precisely modeled. Using well-designed topology learning modules and topology training, we infer the birth/death and merging/splitting of dynamic groups. With the GST model, the proposed multi-object tracker can naturally facilitate the occlusion problem by treating the occluded object and other in-group members as a whole while leveraging overall state transition. Experiments on both RGB and RGB-D datasets confirm that the proposed multi-object tracker improves the state-of-the-arts especially in crowded scenes.**

***Index Terms*—Multi-Object tracking, topology model, grouping**

I. Introduction

Multi-object Tracking (MOT) is a fundamental problem in computer vision, and contributes to many applications including robotics, video surveillance, and intelligent vehicles. While many consider MOT in simple scenes a solved problem, MOT in crowded scenes remains far from being solved when considering complex dynamics and target occlusions. Conventional data association methods [1]–[4] that optimally link target detections with respect to their appearance, motion, and time gap have been intensively investigated; however, modeling the complex dynamics and target occlusions is beyond the scope of their capability.

To model the complex dynamics of moving objects, social behavior analysis [5], [6] has recently been explored. Sociologists find that up to 70% of the pedestrians in a crowd tend to walk in groups. People in the same group are more likely to have similar motion patterns and to be close to each other for better group interaction. This grouping view treats peoples' motion as the result of both their attentions and the interactions with the environment. The group-level MOT methods [7]–[10], as opposed to conventional MOT ones, aim to detect and track groups of objects sharing spatial-temporal characteristics, *i.e.*, velocity, range, and geographical goal. In crowded scenes, however, the truth is that the number and structure of groups vary over time as objects might enter a scene, or disappear at random times. Groups can split, merge, be relatively close to each other or move largely independently on each other, and thus appears complex group dynamics. The group-level MOT methods are competent for initializing, characterizing and tracking groups, but few of them can comprehensively model the group dynamics from the perspectives of dynamic group structures and in-group individual states.

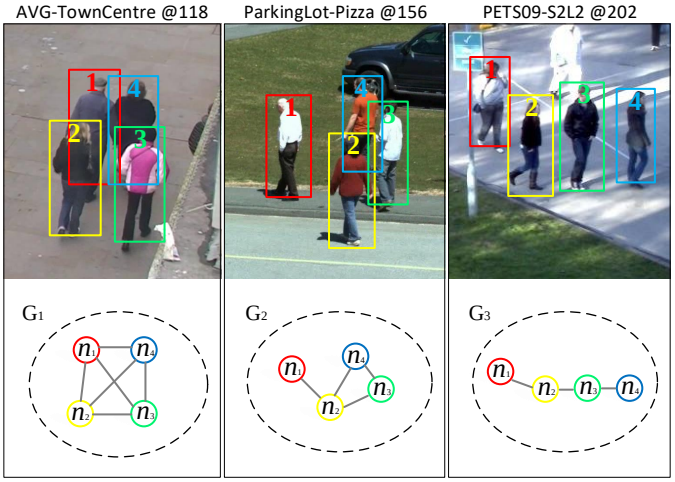


Fig. 1. The motivation of using a Graphical Social Topology (GST) model. The groups G_1, G_2, and G_3 are identical if considered as a node set of indexes, where each node represents one object. But from the view of topology configuration, G_1 contains more edges than G_2 and G_3, which means stronger connection and similar movement among objects. In tracking process, G_2 and G_3 structures are more likely to split than G_1. The GST model with topology changes thus allows propagating more information than a set of indexes. In this paper, we aim to investigate the topology changes in and out groups. which bridge the group and individual tracking.

In this paper, we propose a novel Graphical Social Topology (GST) model to quantify the group dynamics in a graphical way, targeting at tracking the in- and out-group objects accurately. We statistically infer which objects move in formation or have common movement as well as to model behaviors inside groups (split) and between groups (merge). This information fits well with MOT applications where the goal is

Shan Gao, Xiaogang Chen, Xiangyang Ji are with the Department of Automation at Tsinghua University, Beijing, China. (Email: gshan, xiaogangc, xyji@tsinghua.edu.cn)

Qixiang Ye is with the School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, China. (Email: qxye@ucas.ac.cn)

Arjan Kuijper is with Fraunhofer Institute for Computer Graphics Research (IGD) and Technology University of Darmstadt, Germany. (Email: arjan.kuijper@mavc.tu-darmstadt.de)

to differentiate in-group members from out-group objects, or to predict the intention, destination and future manoeuvres of different objects. The motivation for GST (*cf*. Fig. 1) is in the possibility of using the common group information to improve the tracking of individual objects as well as using topology configuration to infer the birth/death and merging/splitting of dynamic groups.

To implement a highly accurate MOT, we formulate a group of objects as a dynamic topological graph. The novelty of this paper is to estimate the group structure jointly with the group objects' states using a topological representation based on the social affinity. With such topology representation, objects are not only assigned to groups, but also connected to each other, which enables the cohesion of a group to be precisely modeled. When a group member is occluded by other members or misdetected by an detector, our model can infer the position using the group relation. To sum up, the main contributions for this work consist of:

- A Graphical Social Topology (GST) model. The social affinity in natural crowds is quantified by the topology graph, this topology relation is formulated as a strong context information to infer the group and individual states.
- A graphical based group learning strategy. The learn strategy integrates birth, update, merge, and split modules to topologize the dynamic groups. Aggregated with the trained typical topology patterns, this strategy facilitates the description of topology transformation.
- A group and individual joint tracking. We contribute a joint framework for the group and individuals tracking, filling in the gap between group modeling and MOT by identifying the group and individual simultaneously.
- RGB and RGB-D applications. We evaluate on both RGB and RGB-D tracking tasks, and demonstrate the proposed method performs favorably against the state-of-the-art methods, especially in crowded scenes.

The remainder of this paper is organized as follows: the related work is described in Section II. The graphical social topology model is introduced in III and the social topology training IV. Section V describes the multiple objects tracking using the proposed graphical social topology model. Experimental results and conclusions are presented in Sections VI and VII respectively.

II. Related Works

We review the most relevant tasks of group and individuals tracking in computer vision, including multiple object tracking, group modeling, and group tracking.

Classical MOT approaches usually explore data association models to link pairwise detections across time frame-by-frame (online) [11]–[16] or in a batch way (global) [1]–[3], [17]–[20]. With the help of optimization algorithms, the data association methods in a global way achieve a higher accuracy than online methods. Zhang *et al*. [1] mapped the Maximum-A-Posteriori framework into a network with a non-overlap constraint on trajectory. Pirsiavash *et al*. [2] proposed a globally-optimal greedy algorithm to search for the successive shortest paths by defining a residual graph in the network. Zamir *et al*. [17] defined a fully-connected graph to connect all the object detections. Meanwhile, A large number of tracking approaches are related to trajectory-level analysis with high-order information. Yang *et al*. [21], [22] used a trajectory-based Conditional Random Field function to learn the affinity and dependency among the object observations online. Wen *et al*. [23], [24] adopted tracklets-dense neighborhoods searching strategy in relation graph to guarantee the trajectory smoothness. Amit *et al*. [25] investigated how to associate the detections by propagating labels on a set of graphs, each graph capturing how either the spatio-temporal or the appearance cues promote the assignment of identical or distinct labels to a pair of detections. Milan *et al*. [18], [26], [27] used energy minimization methods with trajectory-level constraints to distinguish objects' identities. Most of the above researches are based on an assumption that objects move independently, so the models in these researches are mostly designed in aspects of trajectory smoothness and appearance affinity, Social context information among the objects is ignored in these trajectory-level MOT methods.

Social context has been studied intensively in this decade, and it mostly appears as group modeling. Researchers target to a find stable and accurate clustering way to describe movement in the form of groups. Haritaolu *et al*. [28] regarded group detection as a graph partition problem, while Ge *et al*. [29] and Chang *et al*. [30] discovered small groups by bottom-up hierarchical clustering of trajectories based on pairwise objects speed and distance. Another trajectory-based approach was proposed by Zhou *et al*. [31] and it used a coherent filtering algorithm to segment coherent motion in the crowd. Moussaid *et al*. [5] found that group members tend to walk side-by-side at low crowd density and the formation is bent to a V-shape pattern as the density increases. Some work grouped pedestrians by analyzing their relative distances and moving patterns. Li *et al*. [32] used trajectory information of multiple objects to learn models for segmenting different group patterns. These studies provide the trajectory-level analysis to model and discovery groups in crowded and semi-crowed scenes.

These group modeling methods, together with social behavior research, formulated as social force models [5], [33], are used as high-level constraints and have attracted increasing attentions in MOT framework. Pellegrini *et al*. [8] proposed an effective dynamic group model, considering nearby pedestrians' positions. Qin and Shelton [34] used a dual optimization framework and a linear programming solution to model the social group behavior as a high-level clue. Chen *et al*. [7], [35] adopted an online learning strategy to formulate the social behavior as an elementary grouping model. Alahi *et al*. [36] proposed a social affinity map feature to define the motion feature in crowded scenes and utilized it to solve a large-scale pedestrian forecasting problem. Bazzani *et al*. [37] assumed a tight relation of mutual support between the modeling of individuals and groups, promoting the idea that groups are better modeled if individuals are considered and vice versa. These group-based tracking methods confirm that the group model in MOT framework could be a strong and effective

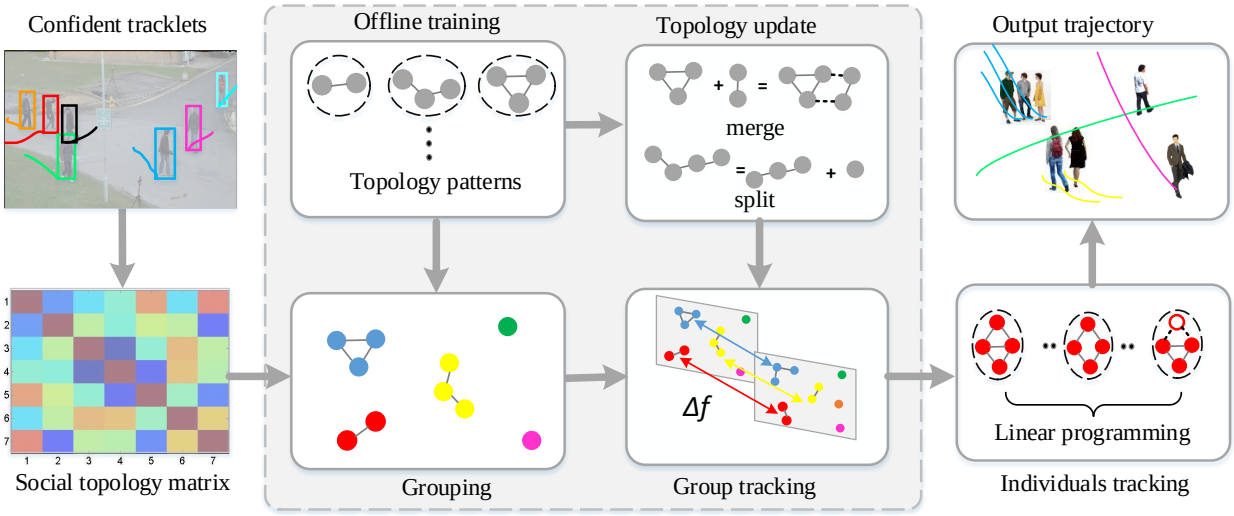


Fig. 2. The framework of the proposed graphical social topology model. First, we group the clusters integrated with trained topology patterns, which is based on the social affinity matrix among confident tracklets. The group tracking is then aggregated with the topology management with the movement of objects, in which a joint group and individual tracking method is proposed to solve the data association problem. Finally, We use the linear programming to solve the individual tracking and output the complete trajectories. Our job mainly focuses on the dashed rectangle how to design and use the effective topological graph to reveal the group and individual dynamics from the view of the social affinity.

constraint. None of them formulates the group changes as a kind of topology transformation. The proposed GST model uses a topology graph to represent the group update, merge, and split flexibly, it is not a pure energy or cost optimization manner solved in a local search manner.

As a comparison with [37]–[39], we focus on utilizing social grouping information as more natural and flexible high-level constraints (*cf.* Fig. 2). We identify groups using a topological graph based on a bottom-up hierarchical grouping approach which starts with individual tracklets as separate clusters and builds a social topology matrix to group the clusters with strong social affinity. Then groups are discovered and updated by leveraging trained topology patterns. Accordingly, our work focuses on using the analytical social groups to maintain individual identities in group and individual tracking.

III. Graphical Social Topology Model

In this section, we present a social topology matrix based on group dynamics and spatial topology. The social topology matrix T encodes the social exchanges occurring among all the objects in a scene. We then introduce two properties of social topology: compactness and consistency. Working with the social topology matrix, these properties are used to regularize the topology of a group in learning and topology training.

A. Social Topology Matrix

At each frame f, the tracklet n_i^f, corresponding to the object i, is represented by a set of state variable $n_i^f = (X_i^f, v_i^f, o_i^f)$, where X_i^f, v_i^f, and o_i^f denote the position, speed, and orientation, respectively. The social affinity matrix $T = \{T_{ij}\}$ measures the social affinity between tracklets n_i and n_j as

$$T = \alpha_d T_d + \alpha_t T_t + \alpha_v T_v + \alpha_o T_o, \tag{1}$$

where T_d, T_t, T_v, and T_o are the social affinities based on distance, time, speed, and orientation at frame f. Here we omit the superscript f for simplicity. α_d, α_t, α_v, and α_o are balance factors.

TABLE I
Notations

Symbol	Description
T	social affinity matrix $\{T_{ij}\}$
n_i	node/tracklet, $n_i = (X_i, v_i, o_i)$
l_i	motion vector of n_i, $l_i = (X_i, v_i)$
G	group $\{G_k\}$
E	edge set in G
N_k	the size of group G_k
g_k	the center of group G_k
π_i	topological representation of n_i in group, $\pi_i = (r_i, \theta_i)$
$D_{n_i}^{G_k}$	degree of node n_i in G_k
C	sampling matrix in individuals
B	sampling matrix in group
L_{ij}	co-existing period between n_i and n_j
A_{ij}	appearance affinity between n_i and n_j
Ψ_{ij}	location affinity between n_i and n_j
M_{ij}	binary association matrix
α	balance factor (Eq. 1)
λ	distance threshold (Eq. 2)
l	co-existing period threshold (Eq. 3)
τ	edge weight threshold(Sec. III-D)

Distance. Pedestrians tend to unconsciously organize the space around them in particular configuration with different degrees of intimacy. The shorter the distance between two persons, the higher the degree of intimacy. We adopt different distance measure strategies according to the states of objects in datasets. In RGB-D datasets with the real-world depth information, d_{ij} denotes the world-coordinate distance between two objects. While in RGB datasets, d_{ij} denotes the distance between the center points of objects' bounding box in images. We set λ as the distance threshold in RGB datasets, which is a learned distance threshold according to the density of the

crowds in training datasets (detailed in Sec. IV). The distance affinity is defined as

$$T_d(n_i, n_j) = \frac{\lambda}{2d_{ij}}. \tag{2}$$

In the computing, we define a limit distance $\lambda = w_i + w_j$ in RGB datasets (w_i and w_j are the width of bounding boxes of tracklets n_i and n_j), beyond which two individuals can be considered not to be interacting with high probability.

Time. The time term indicates how long two tracklets n_i and n_j perform a similar motion and stay close to each other. Members in the same group always appear and disappear at the similar time. Let L_{ij} denote the length of the co-existing period, when the distance d_{ij} between n_i and n_j satisfies $d_{ij} < 2\lambda$. We set this "co-existing period" lasting for at least l frames, as

$$T_t(n_i, n_j) = \frac{L_{ij} + l}{2L_{ij}}. \tag{3}$$

Speed. Objects in a group tend to have the same speed. Let v_i and v_j denote the speeds of tracklets n_i and n_j. The speed affinity between n_i and n_j are defined as

$$T_v(n_i, n_j) = \mathcal{N}(\|v_i - v_j\|), \tag{4}$$

where $\mathcal{N}(\cdot)$ is a min-max normalization operator applied independently for each pairwise tracklets to linearly scale their speed differences into the range $[0, 1]$.

Orientation. We adopt an improved Potts model similar to [40] to define the affinity among different moving orientations as

$$T_o(n_i, n_j) = \frac{1 + \cos(o_i - o_j)}{2}, \tag{5}$$

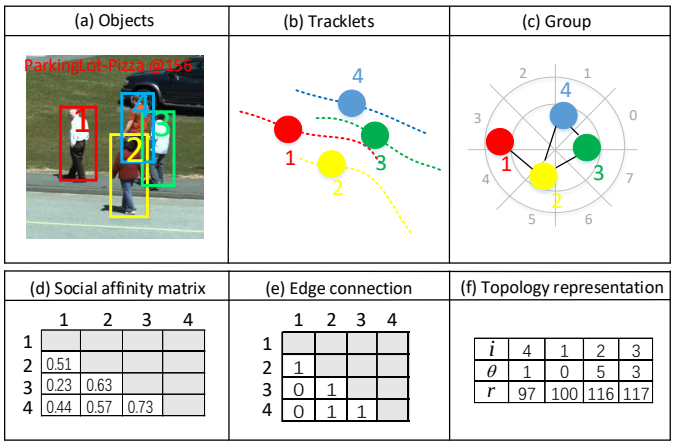
where $o_b = \frac{2\pi q_b}{q}$, and $b = i, j$. The object's moving orientation is divided into q bins. Here we use $q = 9$, which means a resolution of $45°$ from '0' to '8' between neighbor orientations, and '0' means that the object keeps still in continuous frames. q_b is the moving orientation.

Instead of considering the speed and orientation together as in [29], [35], we calculate these two factors in two terms, which make the social topology model applicable in both RGB and RGB-D datasets and enables it effective against the poor detections. Especially, the '0' bin in orientation term is assigned to the stationary object, which makes the stationary pairwise tracklets keep a stable social affinity.

The social topology matrix can be employed as a tool to describe the group dynamics for different applications. The truth is the following graphical social topology model is independent of the social affinity matrix.

B. Topological representation for group

Given a social topology affinity matrix T, a graph is defined as $G = (\{n_1, \cdots, n_N\}, E(T_{ij}))$, wherein N objects constituting the set of nodes $\{n_1, \cdots, n_N\}$ linked by edge set E, and each node n_i is associated with the tracklet of one object (one tracklet equals one node in graph). Using this definition, the Groups G_1, G_2 and G_3 in Fig. 1 can be denoted as $G_1 = (\{n_1, n_2, n_3, n_4\}, \{T_{12}, T_{13}, T_{14}, T_{23}, T_{24}, T_{34}\})$, $G_2 = (\{n_1, n_2, n_3, n_4\}, \{T_{12}, T_{23}, T_{24}, T_{34}\})$ and $G_3 = (\{n_1, n_2, n_3, n_4\}, \{T_{12}, T_{23}, T_{34}\})$. As the weight of edge set E is measured by the dynamic social topology affinity matrix T (Eq. 1), G is a dynamic topological graph representing the structure of in-group members. Fig. 3 visualizes objects in different measurements.



	1	2	3	4
1				
2	0.51			
3	0.23	0.63		
4	0.44	0.57	0.73	

	1	2	3	4
1				
2	1			
3	0	1		
4	0	1	1	

i	4	1	2	3
θ	1	0	5	3
r	97	100	116	117

Fig. 3. Representation of objects in different measurements. (a) objects in bounding boxes; (b) objects in tracklets; (c) objects in a topological group; (d) social affinity matrix among objects in a group; (e) edge connection among objects in a group; (f) topological description of objects.

In [37], group G represents a set of group labels for each object. For example, with five objects, $G = \begin{bmatrix} 1 & 1 & 2 & 2 & 2 \end{bmatrix}$ means that objects 1 and 2 are in group 1 and objects 3, 4 and 5 are in group 2. With the affinity matrix T, G is flexible to update the group structure and model the interaction between group members.

C. Social Topology Property

We introduce two properties, *i.e.*, *compactness* and *consistency* of a social topology. The compactness property quantifies the spatial structure of the topology. The consistency property describes the temporal and spatial evolvement of the in-topology members. Such properties enable the social topology competent to handle group management, such as splitting and merging.

Compactness. We construct a graph among objects and measure the edge density by the total degree as group compactness. In graph theory, degree of a node in a graph is the number of edges connected with the node. D_{n_i} records the number of edges connected with n_i, and D^{G_k} records the total degree of all the nodes in G_k. The compactness constraints are defined as

$$\begin{cases} I : D^{G_k} > 2(N_k - 1), \\ II : \max D_{n_i} = N_k - 1, \end{cases} \tag{6}$$

where N_k is the size of a graph. Constraint I promises that the topology has a high edge density. A qualified topology should have enough edges among nodes. This can exclude the group with a 'line-like' topology in Fig. 1. Constraint II guarantees a tight structure. This enables the in-group members to distribute around a center member, *i.e.*, a 'star-like' topology. The topology of a group should satisfy at least one of the two tightness constraints.

Consistency. We define topology consistency to represent in-group spatial evolvement in sequences. Considering the problem of tracking the motion of groups of targets, each node n_i is characterized by its motion vector $l_i = (x_i; \dot{x}_i; y_i; \dot{y}_i)'$ ($X_i = (x_i; y_i)$ and $v_i = (\dot{x}_i; \dot{y}_i)$); Each node is associated with as well as the target states corresponding variance matrix P_i^0. In two dimensions[1], the state of the ith target is given by:

$$l_i^f = C l_i^{f-F} + \Gamma \varphi^{f-F}, \tag{7}$$

where $C = diag(C_1, C_1)$, $C_1 = \begin{pmatrix} 1 & F \\ 0 & 1 \end{pmatrix}$, $\Gamma = \begin{pmatrix} F/2 & 1 & 0 & 0 \\ 0 & 0 & F/2 & 1 \end{pmatrix}'$, F is the sampling interval and $f - F$ is the system dynamics noise. Considering the nonlinear motion, especially the abrupt speed and orientation changing in the tracking, the system dynamics noise φ is represented as a sum of two Gaussian components $p(\varphi^{f-F}) = \eta \mathcal{G}(0, Q_1) + (1-\eta)\mathcal{G}(0, Q_2)$, where $Q_1 = diag(\delta^2, \delta_1^2)$ and $Q_2 = diag(\delta^2, \delta_2^2)$. δ is a standard deviation assumed constant for x and y, δ_1 and δ_2 control the abrupt changing in x and y orientation.

When modeling the group consistency, the interaction between objects in each group should be considered. In order to describe the group in a whole moving unity, we record a virtual group center $g_k = \frac{1}{N_k} \sum X_i$, where n_i belongs to g_k and N_k is defined as the number of targets in G_k. The center and covariance matrix of each group can be characterized differently, *e.g.*, based on a mixture of Gaussian components. We represent the topology of a group as $\pi_i = (r_i, \theta_i)$, where $r_i = \sqrt{x_i^2 + y_i^2}$ mean the distance between a node and the virtual center. $\theta_i = tan^{-1} \frac{y_i}{x_i}$ means the topology angel in a group, which is quantified in the $[0, 7]$ shown in Fig. 3c and 3f. When updating the group movement, we have the following equation

$$g_k^f = g_k^{f-F} + \sum_{n_i \in g_k}^{N_k} (B l_i^f) + \Gamma \varphi^f, \tag{8}$$

where $B = diag(B_1, B_1)$, and $B_1 = \begin{pmatrix} 0 & F/N_k \\ 0 & 0 \end{pmatrix}$.

D. Group learning

One of the challenges in group modeling is how to update the topology structure and model the interactions among objects. To this end, adding objects to the groups, removing others, splitting, and merging groups are of primary importance. We formulate the topology learning process as four submodules: Birth, Update, Merge, and Split Modules. Algorithm 1 summarizes the details of each module. Compared with the other clustering and inference methods, the proposed model is able to automatically discover the number of groups and construct updating, merging, and splitting events by the dynamic social topology.

[1]In RGB-D datasets, the depth data z and the speed $\dot{z}$ are used as $l_i = (x_i; \dot{x}_i; y_i; \dot{y}_i; z_i; \dot{z}_i)'$, $X_i = (x_i; y_i; z_i)$, $v_i = (\dot{x}_i; \dot{y}_i; \dot{z}_i))$, $C = diag(C_1, C_1, C_1)$, $C_1 = \begin{pmatrix} 1 & F \\ 0 & 1 \end{pmatrix}$, $\Gamma = diag(\Gamma_1, \Gamma_1, \Gamma_1)'$, and $\Gamma_1 = \begin{pmatrix} F/2 & 1 \\ 0 & 0 \end{pmatrix}$.

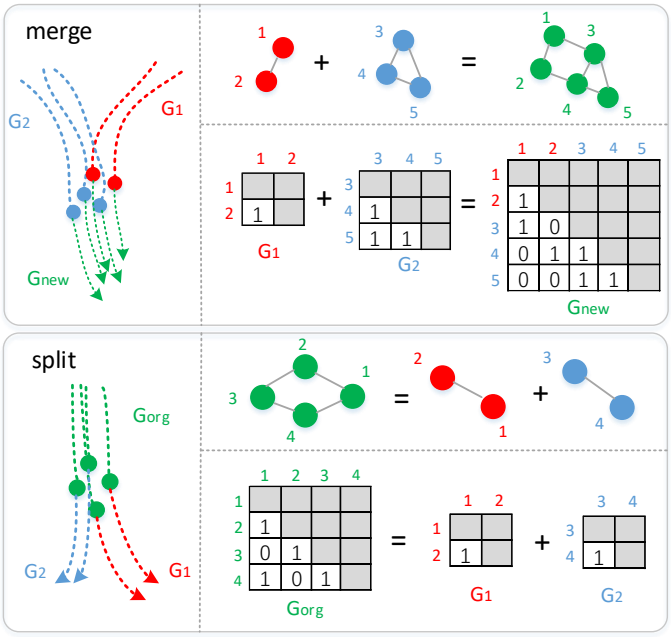


Fig. 4. The illumination of merging and splitting modules in group management. the essence of the new edge is determined by the social affinity T_{ij} and the threshold τ.

Birth. Initially, the edge is an empty set $E = \{\emptyset\}$, and the social affinity T_{ij} is used as the edge weight. There is an edge between n_i and n_j if $T_{ij} < \tau$, τ is the edge threshold trained in Sec. IV. We then group the connected nodes according to 8 kinds of typical topology patterns, shown in Fig. 7, which are offline learned (*cf.* Sec. IV). If the topology of connected nodes matches any typical topology pattern, we divide them as one group. In Birth-Module, we do not initialize groups with a large size. This is able to avoid the large-size topology with false tight structure, which is easy to split in the following frames. In fact, small groups are assembled in Merge-Module when they perform a high social affinity.

Update. Existing edges should be updated at each frame since the graph structure is related to the dynamic spatial configuration. The topology structure of a group is related to the social affinity in tracking, so the existing edges should be updated by social affinity T_{ij}. Further, when T_{ij} between two nodes from different groups less than edge threshold τ, the management goes to the Merge Module. When T_{ij} between in-group members is larger than edge threshold τ, the management goes to the Split Module.

Merge. When two groups move close to each other and last for l frames, they are merged into a big group. However, not all the groups moving close ($T_{ij} < \tau$) can be merged together. First, the new group should satisfy the compactness constraint defined in Eq. 6. Then, the topology of the new group is expected as tight as possible. The total degree of the merged group is $D = D_{G1} + D_{G2} + D_{new}$, where D_{new} is new degrees generated by new edges. We define the merging constraint as

$$D_{new} > \min(N_1, N_2), \tag{9}$$

where N_1 and N_2 are sizes of G_1 and G_2. D_{new} should be

Algorithm 1: Group learning

• Birth

Input: $E = \{\emptyset\}$
Output: $G = \{G_k\}$
Step-1:
Calculate $T = \{T_{ij}\}$ in Eq. 1;
For each connection T_{ij};
IF $T_{ij} < \tau$, $E = E \cup \{(n_i, n_j)\}$;
End for
Step-2:
Cluster E and generate the coarse group set $\{G_k\}$;
For each group G_k
IF D_k dissatisfies Eq. 6, **GOTO** Split-Module;
Regulate the topology by typical topology patterns in Fig. 7;
End for

• Update

Input: $G^{f-1} = \left\{G_k^{f-1}\right\}$, $T^f = \left\{T_{ij}^f\right\}$
Output: $G^f = \left\{{G_k}^f\right\}$
For each group G_k^{f-1}
Step-1:
Update edge weights by T_{ij}, $i, j \in G_k$;
Step-2:
IF group size $N_k^f < N_k^{f-1}$, ADD virtual nodes;
IF edges connect with other groups, **GOTO** Merge-Module;
IF the compactness D_k dissatisfies Eq. 6, **GOTO** Split-Module;
End for

• Merge

Input: G_{k1}, G_{k2}
Calculate the compactness D_{new} of group $G_{k1} \cup G_{k2}$;
If $D_{new} > \min(N_{k1}, N_{k2})$
Output: $G_{new} = G_{k1} \cup G_{k2}$;
Else
Output: G_{k1} and G_{k2};

• Split

Input: G_k, T
Output: $\left\{G_k^{'}\right\}$
While D_k dissatisfies compactness constraints in Eq. 6
Cut the edge with the lowest T_{ij};
$G_k^{'} = G_{k1} + G_{k2}$, $D_k = D_{k1} + D_{k2} + D_{new}$;
End while

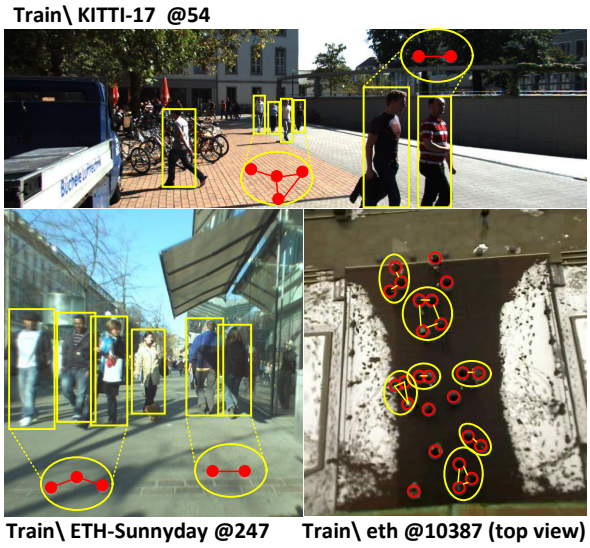


Fig. 5. Learned typical topology pattern samples (marked in red color).

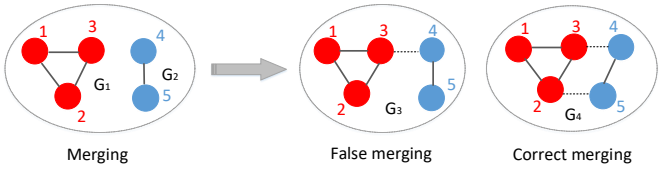


Fig. 6. An example for the false and correct merging issue. G_3 and G_4 are new groups consisting of G_1 and G_2, and the dash line are new edges, but only G_4 satisfies the merging constraint (only the D_{new} of G_4 is large than $\min(N_1, N_2) = 2$) according to Eq. 9.

larger than the size of the smaller group, $\min(N_1, N_2)$. An example is shown in Fig. 6.

Split. When the members inside a group move with different velocities and/or orientations, this group is considered to split into small groups. Under this circumstance, the edge weight will decrease greatly, and that edge will be cut when T_{ij} is less than the threshold τ.

The above group management manner has the following advantages. First, one can redesign and debug individual components effectively. Secondly, the modular framework makes it easy to replace each module or to insert a new one. Third, it enables training each module separately, speeding up the management process in joint training.

IV. Topology Pattern Training

In this section, we investigate the spatial organization of the walking pedestrian topology to determine whether there is any specific pattern of spatial topological configuration. These typical topology patterns are used for the Birth-module in group learning. In addition, two parameters in our model need to be trained: the edge threshold τ (the threshold of T_{ij}) and distance threshold λ (the threshold of social distance d_{ij}).

In training sequences, given a set of detections and the corresponding ground truth (GT) object annotations, the GT objects' IDs are first assigned to each detection as complete trajectories. Supposing there are N objects (trajectories) in one frame and their co-existing time are longer than l frames, we calculate the $N \times N$ social topology matrix T according to Eqs. 1-5 and get a fully connected graph, where the edge weight is the social topology affinity T_{ij}. In clips of the sequences (which have groups of people), we give an augmented τ value to cut the edge connection in the graph and obtain different grouping results. We then calculate the number of groups and how many times of object switching among groups compared with GT. We search the optimal τ with lowest group amount error ε_g and switching error ε_s as $\underset{\tau}{argmin}\, \varepsilon_g + \varepsilon_s$. The λ is the maximum distance between in-group members when we use this optimal τ.

In RGB datasets, we use average width w of objects' bounding boxes, the parameter of λ is the average radius r, measured by pixel. In RGB-D datasets, we use the world-coordinate distance (meter) between objects' center. Fig. 5 shows the edge connections in different topology patterns in training. When $d_{ij} < 2\lambda$, the social affinity T_{ij} is calculated, and when $T_{ij} > \tau$, there exists an edge between two corresponding nodes in the topology graph. Fig. 7 summarizes 8 kinds of typical topology patterns, which are the most common configurations in the size from 2 to 4.

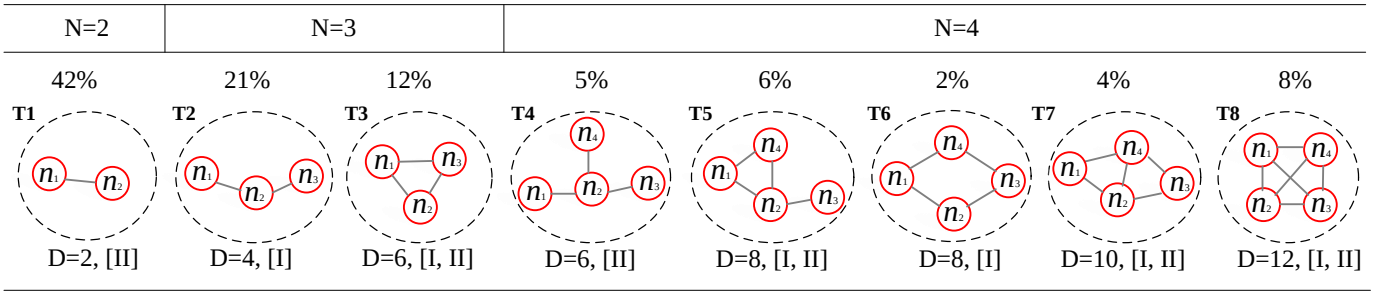


Fig. 7. The learned basic topology patterns. N and D record the size and total degree of a topology pattern, [I,II] denotes which compactness constraint the topology pattern satisfies in Eq. 6. The frequency of each topology pattern in the train datasets is shown above each pattern.

V. Tracking with Graphical Topological Model

In this section, we present a bottom-to-top MOT method based on the proposed graphical social topology model. We described how to manage the group in the group management part, but how to identify the same group in the sequence and how to associate the individuals are still unsolved. The mathematical formulation of MOT is given as follows. Suppose that a set of tracklets $\mathcal{L} = (n_1, \cdots, n_n)$ is generated from a video sequence. A tracklet n_i is a consecutive sequence of detection responses or interpolated responses that contain the same object. The goal is to associate tracklets that correspond to the same objects, given certain spatial-temporal constraints.

A. Group Tracking

The motivation of using the group tracking in the MOT method because that the group, in fact, is easily tracked than the individuals. The whole group usually occupies a large region than a single object. This also means groups would not get lost or drift. The group configurations of continuous frames are highly correlated due to the temporal smoothness of group-member's trajectories. This observation is exploited in our framework, where the grouping configuration at one time step can be used as a reference for the next.

Our tracking method adopts the off-line strategy in the association process. A set of tracklets is generated after low-level association in an overlapping manner, but only confident tracklets are considered for grouping analysis, as there might be false alarms and incorrect associations in the input tracklets. Based on the observation that inaccurate tracklets are often short, we define a tracklet as the confident one if it is long enough. We then cluster the confident tracklets co-occurring at the same time, into different groups based on the social topology matrix T among and utilize the learned edge threshold τ in train datasets to decide whether there is an edge between pairwise nodes.

If the clustering is perfect and people move in the same configuration in the whole sequence, it would suffice to link the groups instead of the tracklets. However, in the practical setting, the grouping is not perfect and people break away from groups. Some grouping results are interrupted by the poor detection inputs. Hence, we link the groups of the same topology through the time span of the "poor detection". Recall that a group is modeled as a set of nodes and edges in a graph G_k. We use the consistency of the topology property to estimate the virtual center states of groups.

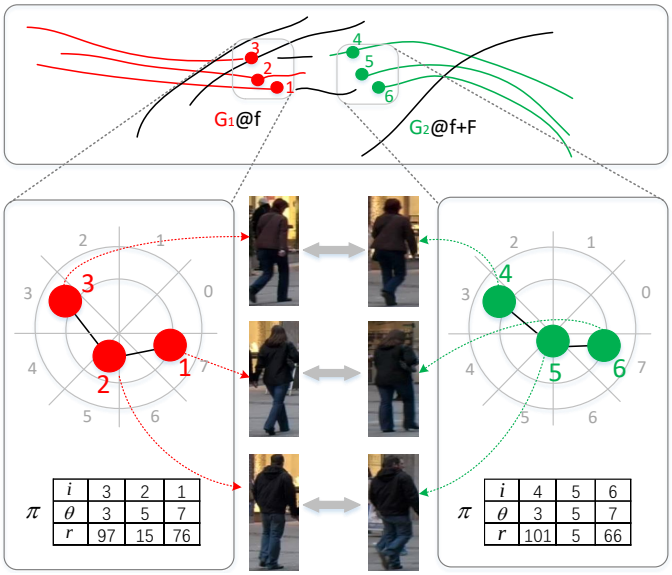


Fig. 8. Illustrations of the group tracking and individuals tracking. Group G_1 and G_2 in frames f and $f+F$ are linked by the group states estimate function Eq. 10. Individuals in the group are associated according to the topology configurations $\pi_i = (\theta_i, r_i)$ and appearance affinity in the grouping context.

$$\hat{g}_k^{f+F} = g_k^f + \sum_{n_i \in G_k}^{N} (Bl_i^f) + \Gamma\varphi^f, \tag{10}$$

where the definition of parameters B and Γ are the same with those in Eq. 8, and F is the time interval between two groups in sequences.

B. Individual Tracking

After got the group tracking results, individuals in the group also should be identified. Given the group relation $G = (\{n_1, \cdots, n_N\}, E)$, we adopt a Linear Programming framework [4], [41] to solve the in-group association problem. Compared with the global group association, the in-group matching is a subgraph searching problem in the grouping context. The beginning and end time of the individuals is equal to the life-span of the group.

$$\arg\max_X \sum_{i,j} (A_{ij} + \Psi_{ij}) M_{ij}, \tag{11}$$

where $\sum_i M_{ij} \leq 1$ and $\sum_j M_{ij} \leq 1$. M_{ij} is a binary indicator matrix in group G_k, which decides whether the tracklets n_i and n_j belong to the same object. A_{ij} denotes the appearance affinity between nodes n_i and n_j, and here, a concatenation of HSV color histogram and HOG features is used as the feature descriptor. Ψ_{ij} denotes the position affinity ($\pi_i = (r_i, \theta_i)$) according to the group center g_k. We encode the angle deviation θ_i referring to the topology center as a Normal Distribution. The Linear Programming problem in Eq. 11 can be solved by Hungarian algorithm [41] and Iterative Approximation method [4] or interpreted by a network flow method and solved by the successive shortest paths [2], [3].

C. Joint Group-individual tracking

Similar with other tracking-by-detection methods, the proposed model also relies on a fixed set of detections as input. This has the drawback that much of the image information is discarded during the non-maxima suppression step built into any detector, potentially ignoring semi-occluded objects. The in-group members often occlude each other.

In order to address this issue, we add a virtual node to the group a group if one frame does not contain any appropriate detection, the virtual node is estimated. Multiple objects moving together usually causes occlusion, particularly, the objects are partially or totally occluded. The best detector is even not able to discover the object without context information. In this case, group management will treat two groups in different size as different groups. Nevertheless, the centers of groups could be linked by the Eq. 10. We consider adding virtual nodes to the group with short members to match the group and infer the positions of the occluded objects by the topology configurations. Virtual nodes $\hat{n}_i$ are added to such a group when the size of the group is less than that should be. The spatial positions of the virtual nodes are estimated as

$$\hat{l}_o^f = l_o^{f-F} + \sum_{n_i \in G_k}^{N} (B l_i^f) + \Gamma \varphi^f, \tag{12}$$

where F is the occluding time of the estimated object l_o, and l_i denotes the motion vector of other members in group G_k.

VI. Implementations and Experiments

In this section, we provide some details of the proposed GST model, particularly the parameters used in different parts. We then evaluate the proposed multi-target tracking algorithm, as well as comparing it with recent state-of-the-art methods. Experimental results clearly show the benefits of utilizing social topology in multi-target tracking.

A. Implementation details

Table II shows two sets of parameters in the proposed GST model. They are slightly different according to the observations in RGB and RGB-D datasets. We assign two values to λ in terms of the density of crowds. The low density means that the number of pedestrians is lower than 20.

Single object consistency. When objects do not move in group, these tracklets need to be connected as a long trajectory without the grouping information. We modify the objective function in Eq. 11 as $\arg\max_X \sum_{i,j} (A_{ij} + \Psi_{ij} + T'_{ij}) M_{ij}$. Here, we drop the position affinity Ψ_{ij} in Eq. 11, instead adding the estimated position affinity Ψ_{ij} and motion affinity T'_{ij}. The affinity Ψ_{ij} is calculated between the start position of tracklet n_i with the estimated position of $\hat{n}_i$ as $\Psi_{ij} = \mathcal{G}\left(X_i - \hat{X}_i, \sum_X\right) \mathcal{G}\left(X_j - \hat{X}_j, \sum_X\right)$, where $\hat{X}_i$ and $\hat{X}_j$ are estimated by the individual consistency property in Eq. 7. $\mathcal{G}(\cdot)$ is the Gaussian function ranging in $[0, 1]$. The motion affinity in Eq. 1 is modified as $T'_{ij} = T_v(n_i, n_j) + \alpha'_o T_o(n_i, n_j)$.

Speed and orientation estimation. The speed and orientation of each tracklet are calculated according to the change of its position in continuous frames. In RGB-D datasets, the real distance of objects in continuous frames are incorporated to formulate the walking pace and orientation. However, in RGB datasets, bounding boxes of objects are not always accurate, particularly in crowd scenes. Inspired by the image-based motion descriptors [20], [42], we design a Local Motion Descriptor to encode the relative motion pattern between two bounding boxes in continuous frames, which combines the Local key Points Detector [43] and the Optical Flow Algorithm [44].

TABLE II
Trained Parameters

Parameter	α_d	$\alpha_{t,v,o}$	l	τ	λ (low)	λ (high)
RGB	0.4	0.2	5	0.5	25pixels	18pixels
RGB-D	0.4	0.2	5	0.5	2meters	1.5meters

TABLE III
Datasets for training and test

Task	Type	Dataset
Training	RGB	MOT benchmark training sequence
	RGB-D	SDL-Crossing
Test	RGB	MOT benchmark test sequence
	RGB-D	Sync, SDL-Campus, SDL, LIPD

B. Dataset and Metrics

The proposed methods are tested and evaluated on two kinds of publicly available datasets, RGB and RGB-D datasets, which are summarized in Table III. We compare the proposed GST model with the state-of-the-art trackers: DP [2], SSP [3], DCO-X [18], SegTrack [20], MotiCon [15], MDP [14], and SCEA [16].

We adopt the commonly used CLEAR MOT tracking metrics [45], [46] to evaluate the tracking performance. Recall and Precision (Prec.) are two basic metrics. Multi-Object Tracking Accuracy (MOTA), Multi-Object Tracking Precision (MOTP), Mostly Tracked (MT), Partially Lost (PL), and Mostly Lost (ML) scores are computed on the entire trajectories and measure how many Ground Truth trajectories (GT) are successfully tracked (tracked for at least 80%), partial tracked (tracked from 20% to 80%) and lost (tracked for less than 20%). Fragment (Frag.) and IDS record how many times the ground truth trajectory is interrupted and switched by a false ID. FP and FN record the number of false positives and negatives (missed objects), respectively.

TABLE IV
COMPARISONS OF TRACKING RESULTS ON TWO RGB DATASETS IN MULTIPLE OBJECTS CHALLENGE BENCHMARK [45]. FOR THE ITEMS WITH ↑, HIGHER SCORES INDICATE BETTER RESULTS, FOR THOSE WITH ↓, LOWER SCORES INDICATE BETTER RESULTS.

Dataset	Method	MOTA ↑	MOTP ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓	Frag. ↓
MOT Benchmark	DP [2]	14.5±13.9	70.8	6.0%	40.8%	13,171	34,814	4,537	3,090
	DCO-X [18]	19.6±14.1	71.4	5.1%	54.9%	10,652	38,232	**521**	819
	SegTrack [20]	22.5±15.2	**71.7**	5.8%	63.9%	**7,890**	39,020	697	**737**
	MotiCon [15]	23.1±16.4	70.9	4.7%	52.0%	10,404	35,844	1,018	1,061
	MDP [14]	30.3±14.6	71.3	**13.0%**	38.4%	9,717	32,422	680	1,500
	SCEA [16]	29.1±12.2	71.1	8.9%	47.3%	6,060	36,912	604	1,182
	GST	**33.5±13.9**	71.3	12.0%	**35.3%**	9,468	**31,963**	766	1,394

TABLE V
COMPARISONS OF TRACKING RESULTS ON AVG-TOWNCENTRE AND PETS09-S2L2 SEQUENCES IN MOT BENCHMARK.

Dataset	Method	MOTA ↑	MOTP ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓	Frag. ↓
AVG-TownCentre	DP [2]	6.6	69.4	4.4%	35.8%	876	4,482	1,317	562
	SegTrack [20]	3.3	69.3	0.9%	86.3%	**235**	6,528	151	108
	MotiCon [15]	11.9	**70.3**	0.9%	69.9%	353	5,872	**74**	**75**
	MDP [14]	25.4	69.7	17.7%	33.6%	1,517	**3,691**	122	264
	SCEA [16]	29.3	69.6	15.0%	42.9%	738	4,226	88	233
	GST	**33.9**	70.2	**22.1%**	**30.1%**	942	3,756	113	163
PETS09-S2L2	DP [2]	33.8	69.4	7.1%	**9.5%**	948	4,410	1,029	705
	DCO-X [18]	37.5	**70.7**	4.8%	16.7%	638	5,200	189	209
	MotiCon [15]	46.6	67.6	9.5%	14.3%	560	4,354	238	264
	SegTrack [20]	46.1	70.6	**26.2%**	16.7%	1,213	**3,773**	211	211
	SCEA [16]	44.6	69.3	7.1%	14.3%	**536**	4,633	175	289
	GST	**51.1**	70.5	16.8%	12.1%	783	3,964	**191**	**187**

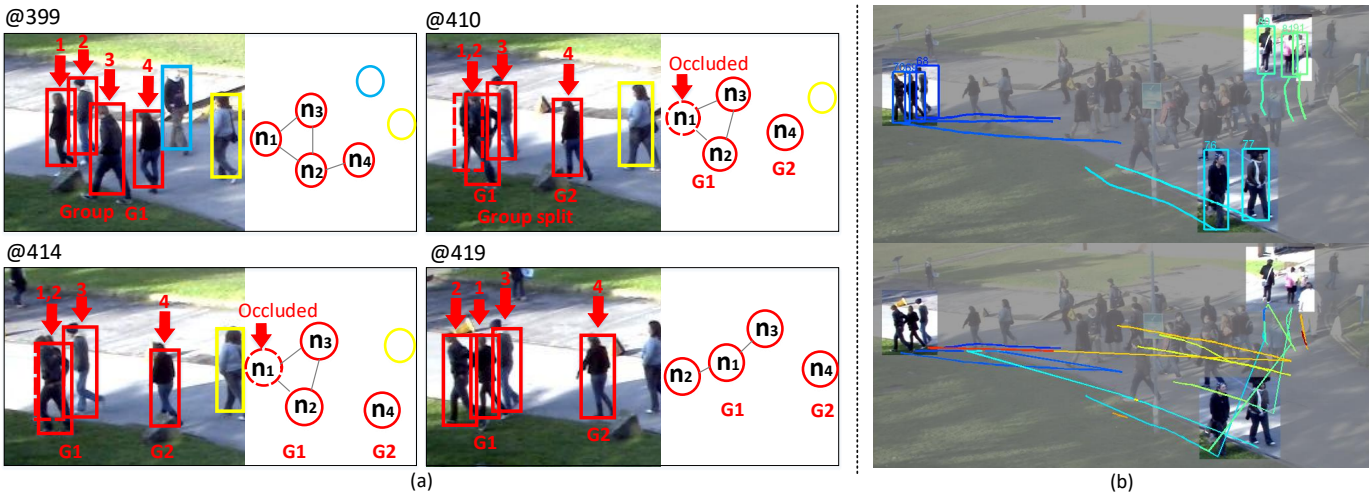


Fig. 9. The performance improvement from topological grouping in MOT benchmark PETS09-S2L2 sequence. We choose several typical groups in the sequence. (a) In-group occlusion issue solved by the topological estimation. (b) Results comparison between GST and "without grouping" (here, we treat all the objects as individuals.

C. Evaluation on RGB datasets

We first apply the proposed model to the MOT video sequences [45]. MOT benchmark is composed of 11,286 frames (∼16.5 minutes) with varying FPS. The dataset is composed of 11 training and 11 test video sequences. Some of the videos are recorded using a mobile platform and the others are from surveillance videos. As it is composed of videos with various configurations, tracking algorithms tuned for a specific scenario would not work well in general. Here we use the same edge threshold $\tau = 0.5$ for all the test sequences, which is learned in training sequences of the MOT benchmark (*cf.* Table II). To keep consistent with previously reported numbers, we follow the exact same evaluation protocol as all other approaches [2], [14], [15], [20], [27], and use their reported results on MOT website. We use public detections [45] for evaluation.

Table IV summarizes the accuracy of the proposed method GST and other state-of-the-art methods on MOT benchmark. It is observed that our model outperforms other state-of-the-art methods. It achieves the highest accurate metric (MOTA). It respectively achieves >8% and >5% performance gain on the AVG-TownCentre and PETS09-S2L2 sequences. These two sequences are most crowded sequences, which contain much more natural group behaviors than other sequences. Tracking

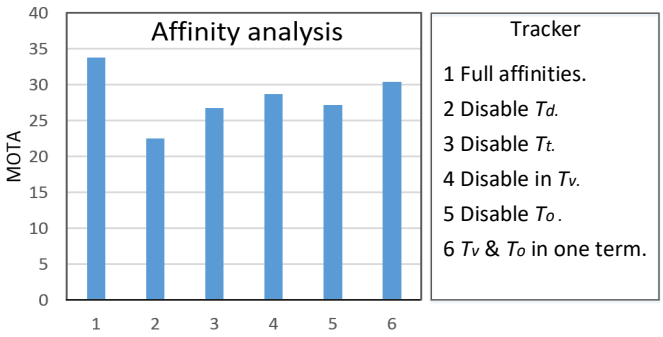


Fig. 10. Social affinity analysis on MOT benchmark.

samples are shown in Fig. 11.

D. Evaluation on RGB-D datasets

We then test the proposed model on three RGB-D datasets: ISR-sync dataset [47], SDL dataset [48], and LIPD dataset [49]. These datasets are recorded from a sensor acquisition system mounted on a moving platform (*e.g.*, instrument-equipped Yamaha vehicle), driving in an urban environment. Each video sequence has a variable number of target objects (Car, Pedestrian, and Cyclist). The videos are recorded at 10 FPS. These datasets are very challenging since 1) the scenes are crowded (occlusion and clutter), 2) the camera is not stationary, and 3) target objects appears in arbitrary location with variable sizes. Many conventional assumptions adopted in MOT with a surveillance camera is not applicable in this case (e.g. fixed entering/exiting location, background modeling, etc). Table VI shows the results of our model compared with the state-of-the-art trackers [3], [20], [27], [48]. GST is our model using only image-level information, and GST+D incorporates the real-coordinate depth data. We use $\tau = 0.5$ as the threshold of the edge weight (*cf.* Table II), which is learned in SDL-Campus dataset. Compared with image based trackers [3], [20], [27], GST+D shows more accurate and robust results (improving $\sim$12% in Recall, $\sim$11% in Precision), at the same time, achieves the almost the highest performance in all the other metrics: MT, PL, ML, IDS and Frag.. Even compared with the depth-based tracker [48], our model improves more than 2% in Recall and Prec. in ISR-Sync and SDL datasets. It is observed that the group initializing, merging, and splitting events happening in tracking, which are shown in the first row of Fig. 11. We also observe that the depth information in GST+D is able to exclude the double detections for one object, which create ghost trajectories in association.

Our own baseline model, GST without using the depth information, also outperforms the other RGB based tracker, improving by $\sim$4% in Recall and Prec. and achieving lower Frag. and IDS errors. Such experiments and comparisons demonstrate that the proposed social topology model is applicable to different kinds of applications, as well as improving the state-of-the-art.

E. Analysis

Contribution of different affinity component. We investigate the contribution of different components in our social affinity matrix by disabling a component at one time and then examining the performance drop in terms of MOTA on the MOT benchmark, which is shown in Fig. 10. We disable the affinity in distance (Eq. 2), time (Eq. 3), speed (Eq. 4) and orientation (Eq. 5) respectively. In addition, we combine the speed and orientation terms in Eq. 11 as one term $T_{v+o} = \mu\mathcal{N}(\|v_i - v_j\|) + (1-\mu)\mathcal{N}(\|o_i - o_j\|)$, where $\mathcal{N}(\cdot)$ is a min-max normalization operator ranging [0, 1]. The other two terms are kept as same as Eqs. 2 and 3. It is observed that new social affinity T decreases the performance of the proposed model. It verifies that formulating speed and orientation in two terms is more robust than considering them together.

Evolvement of the group. Note that the proposed model consistently outperforms all previous methods in tracking accuracy (MOTA). This is mostly due to the graphical topology representing the group dynamics and group learning is able to handle the group birth/death and split/merge in a flexible way, which is shown in Fig. 9. Moreover, our model can estimate the position of the entirely occluded targets in topology and add virtual nodes n_1 according to the previous topology structure in Fig. 9, which allows one to accurately recover more hidden trajectories. Fig. 9b visualizes the tracking results of several groups. compared to the results without using the group context, the proposed GST model is able to provide in-group members a stable topological reference. However, the results without grouping are mixed with other trajectories and some are interrupted by the false detections.

Group Discovery. Group discovery is provided by the group indicator matrix T in the topology model, which indicates the relationship between individuals and groups. In [29], [34], the following group discovery evaluation method is adopted: each pedestrian is coded into one of two categories: alone or in a group. Since we do not have all the annotations on the RGB and RGB-D datasets, we are not able to conduct comparative experiments on all the datasets. In evaluation, we annotate group identity in the PETS09 and AVG-TownCentre sequences. Match rate indicates the percentage of persons that are classified correctly. Compared with existing methods [29], [39], [50], [51], experiments show that our group discovery component produces more reasonable results. In the test datasets, it produces 85% matching rate on more than 300 trajectories. Recall that the same person in different time windows is treated as different persons [29]. In [34], the pedestrians are only divided into alone and pair categories. In contrast, our approach can keep individual and group identity consistent in frames and achieve substantial agreement with human annotator on this dataset. It is also observed that 36% of the people moving in groups in the AVG-TownCentre sequence, it is 65% in the PETS09-S202 sequence.

VII. Conclusion

We proposed the Social Topology Model to solve the multi-object tracking problem in a joint group modeling and MOT framework. The dynamics of moving objects were formulated in a developed graph representation. We learned the typical topology configurations in training datasets and

TABLE VI
COMPARISONS OF TRACKING RESULTS ON THREE RGB-D DATASETS. GST+D DENOTES THE PROPOSED METHOD AND GST IS WITHOUT USING THE DEPTH DATA.

Dataset	Method	Recall ↑	Prec.↑	GT	MT ↑	PL ↓	ML ↓	IDS ↓	Frag.↓
ISR-Sync [47]	SSP [3]	69.6%	72.8%	66	9.0%	66.8%	24.2%	345	323
	DCO-X [18]	73.4%	78.3%	66	19.6%	60.8%	19.6%	**89**	125
	SegTrack [20]	76.2%	79.2%	66	25.8%	57.5%	16.7%	102	147
	DSA [48]	85.0%	89.7%	66	28.8%	57.5%	**13.7%**	90	108
	GST	83.9%	85.4%	66	19.6%	65.3%	15.1%	92	**103**
	GST+D	**87.5%**	**92.3%**	66	**31.8%**	**47.0%**	21.2%	118	134
SDL [48]	SSP [3]	62.9%	70.5%	92	9.8%	59.8%	30.4%	168	189
	DCO-X [18]	70.4%	76.4%	92	19.6%	**55.4%**	25.0%	65	74
	SeqTrack [20]	72.3%	77.8%	92	18.4%	70.8%	10.8%	**55**	71
	DSA [48]	82.4%	87.3%	92	25.0%	59.8%	15.2%	60	68
	GST	79.5%	85.1%	92	19.6%	69.6%	10.8%	61	**60**
	GST+D	**84.4%**	**89.0%**	92	**30.4%**	59.8%	**9.8%**	58	71
LIPD [49]	SSP [3]	72.8%	76.4%	77	10.4%	**55.8%**	33.8%	324	219
	DCO-X [18]	78.4%	78.6%	77	19.5%	58.4%	22.1%	92	123
	SegTrack [20]	77.6%	80.2%	77	13.0%	67.5%	19.5%	75	118
	GST	82.3%	86.6%	77	20.2%	60.3%	19.5%	86	**62**
	GST+D	**86.7%**	**90.0%**	77	**33.8%**	**55.8%**	**10.4%**	**71**	65

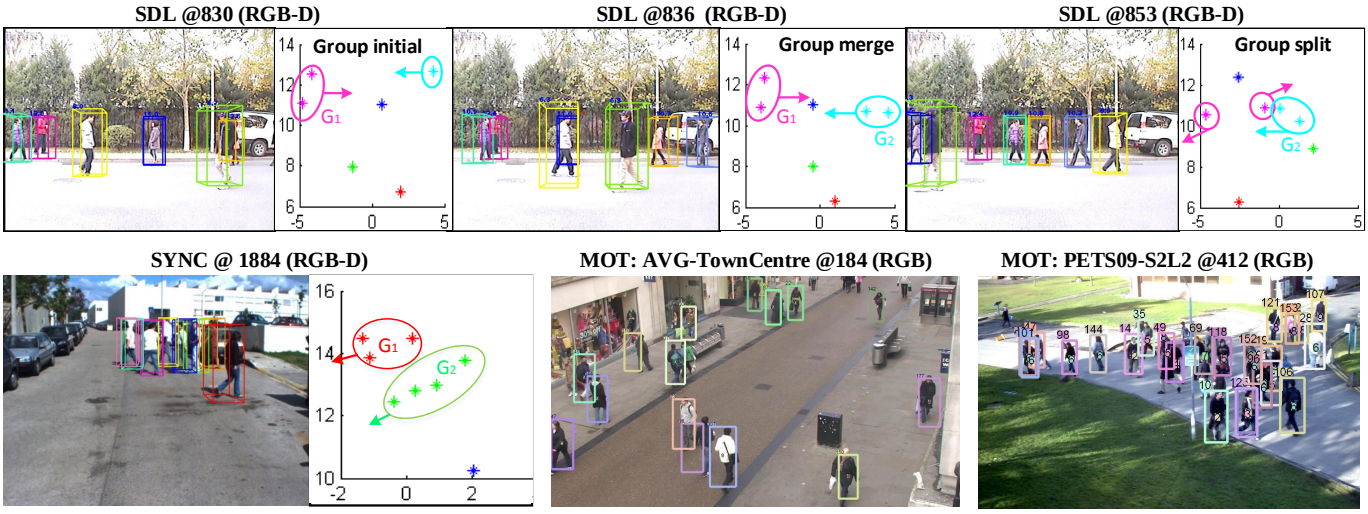


Fig. 11. Tracking results of our approach. First row shows the tracking results of SDL datasets, which contain the group birth, split and merge events. We visualize the results in depth domain of RGB-D datasets. See more results in supplementary materials.

implemented these trained topology patterns to infer the group structure and dynamics combining with a social topology matrix. Meanwhile, we solved the self-occlusion problem in the topology update and identified the individual objects after grouping. Experiments on both RGB-D and RGB datasets showed that our topology based multi-object tracking approach significantly improved the MOT performance, validating that group-level constraint is effective for tracking in crowd scenes.

REFERENCES

[1] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[2] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1201–1208.

[3] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, "Multiple object tracking using k-shortest paths optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1806–1819, 2011.

[4] R. T. Collins, "Multitarget data association with higher-order motion models," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1744–1751.

[5] M. Moussaïd, N. Perozo, S. Garnier, D. Helbing, and G. Theraulaz, "The walking behaviour of pedestrian social groups and its impact on crowd dynamics," *PloS one*, vol. 5, no. 4, p. e10047, 2010.

[6] H. Singh, R. Arter, L. Dodd, P. Langston, E. Lester, and J. Drury, "Modelling subgroup behaviour in crowd dynamics dem simulation," *Applied Mathematical Modelling*, vol. 33, no. 12, pp. 4408–4423, 2009.

[7] X. Chen, Z. Qin, L. An, and B. Bhanu, "Multi-person tracking by online learned grouping model with non-linear motion context," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 12, pp. 2226–2239, 2016.

[8] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *IEEE International Conference on Computer Vision*, 2009, pp. 261–268.

[9] Z. Qin and C. R. Shelton, "Improving multi-target tracking via social grouping," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1972–1978.

[10] L. Leal-Taixé, G. Pons-Moll, and B. Rosenhahn, "Everybody needs somebody: Modeling social and grouping behavior on a linear programming multi-people tracker," in *IEEE International Conference on Computer Vision Workshops*, 2011, pp. 120–127.

[11] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, "Robust multiperson

tracking from a mobile platform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 10, pp. 1831–1846, 2009.

[12] W. Choi, C. Pantofaru, and S. Savarese, "A general framework for tracking multiple people from a moving camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 7, pp. 1577–1591, 2013.

[13] Z. Khan, T. Balch, and F. Dellaert, "Mcmc-based particle filtering for tracking a variable number of interacting targets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1805–1819, 2005.

[14] Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: Online multi-object tracking by decision making," in *IEEE International Conference on Computer Vision*, 2015, pp. 4705–4713.

[15] A. Milan, L. Leal-Taix, K. Schindler, and I. Reid, "Joint tracking and segmentation of multiple targets," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5397–5406.

[16] J. H. Yoon, C.-R. Lee, M.-H. Yang, and K.-J. Yoon, "Online multi-object tracking via structural constraint event aggregation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1392–1400.

[17] A. R. Zamir, A. Dehghan, and M. Shah, "Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs," in *European Conference on Computer Vision*, 2012, pp. 343–356.

[18] A. Milan, K. Schindler, and S. Roth, "Multi-target tracking by discrete-continuous energy minimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2054–2068, 2016.

[19] A. Dehghan, S. Modiri Assari, and M. Shah, "Gmmcp tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4091–4099.

[20] L. Leal-Taixé, M. Fenzi, A. Kuznetsova, B. Rosenhahn, and S. Savarese, "Learning an image-based motion context for multiple people tracking," in *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 2014, pp. 3542–3549.

[21] B. Yang, C. Huang, and R. Nevatia, "Learning affinities and dependencies for multi-target tracking using a crf model," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1233–1240.

[22] B. Yang and R. Nevatia, "An online learned crf model for multi-target tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2034–2041.

[23] L. Wen, W. Li, J. Yan, Z. Lei, D. Yi, and S. Z. Li, "Multiple target tracking based on undirected hierarchical relation hypergraph," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1282–1289.

[24] L. Wen, Z. Lei, S. Lyu, S. Z. Li, and M. H. Yang, "Exploiting hierarchical dense structures on hypergraphs for multi-object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 1983–1996, 2016.

[25] K. K. C. Amit, L. Jacques, and C. D. Vleeschouwer, "Discriminative and efficient label propagation on complementary graphs for multi-object tracking." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 1, pp. 61–74, 2017.

[26] A. Milan, K. Schindler, and S. Roth, "Detection-and trajectory-level exclusion in multiple object tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3682–3689.

[27] A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 58–72, 2014.

[28] H. Ismail and F. Myron, "Detection and tracking of shopping groups in stores," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2001, pp. 1063–6919.

[29] W. Ge, R. T. Collins, and R. B. Ruback, "Vision-based analysis of small groups in pedestrian crowds," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 5, pp. 1003–1016, 2012.

[30] M. C. Chang, N. Krahnstoever, and W. Ge, "Probabilistic group-level motion analysis and scenario recognition," in *IEEE International Conference on Computer Vision*, 2011, pp. 747–754.

[31] B. Zhou, X. Tang, and X. Wang, "Coherent filtering: detecting coherent motions from crowd clutters," in *European Conference on Computer Vision*, 2012, pp. 857–871.

[32] R. Li and R. Chellappa, "Group motion segmentation using a spatio-temporal driving force model," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2038–2045.

[33] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, pp. 4282–4286, 1995.

[34] Z. Qin and C. Shelton, "Social grouping for multi-target tracking and head pose estimation in video." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2082–2095, 2016.

[35] X. Chen, Z. Qin, L. An, and B. Bhanu, "An online learned elementary grouping model for multi-target tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1242–1249.

[36] A. Alahi, V. Ramanathan, and L. Fei-Fei, "Socially-aware large-scale crowd forecasting," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2211–2218.

[37] L. Bazzani, M. Zanotto, M. Cristani, and V. Murino, "Joint individual-group modeling for tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 4, pp. 746–759, 2015.

[38] L. Bazzani, M. Cristani, G. Paggetti, D. Tosato, G. Menegaz, and V. Murino, "Analyzing groups: a social signaling perspective," in *Video Analytics for Business Intelligence*, 2012, pp. 271–305.

[39] A. Gning, L. Mihaylova, S. Maskell, S. K. Pang, and S. Godsill, "Group object structure and state estimation with evolving networks and monte carlo methods," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1383–1396, 2011.

[40] F.-Y. Wu, "The potts model," *Review of Modern Physics*, vol. 54, no. 1, pp. 235–268, 1982.

[41] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1, pp. 83–97, 1955.

[42] W. Choi, "Near-online multi-target tracking with aggregated local flow descriptor," in *IEEE International Conference on Computer Vision*, 2015, pp. 3029–3037.

[43] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *European Conference on Computer Vision*, 2006, pp. 430–443.

[44] G. Farnebäck, "Very high accuracy velocity estimation using orientation tensors, parametric motion, and simultaneous segmentation of the motion field," in *IEEE International Conference on Computer Vision*, 2001, pp. 171–178.

[45] "Multiple object tracking benchmark," http://motchallenge.net.

[46] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: the clear mot metrics," *Journal on Image and Video Processing*, vol. 1, pp. 1–10, 2008.

[47] L. Oliveira, U. Nunes, P. Peixoto, M. Silva, and F. Moita, "Semantic fusion of laser and vision in pedestrian detection," *Pattern Recognition*, vol. 43, no. 10, pp. 3648–3659, 2010.

[48] S. Gao, Z. Han, D. Doermann, and J. Jiao, "Depth structure association for rgb-d multi-target tracking," in *IEEE International Conference on Pattern Recognition*, 2014, pp. 4152–4157.

[49] "Lipd dataset in urban environment," http://www2.isr.uc.pt/ cpremebida/dataset.

[50] I. Chamveha, Y. Sugano, Y. Sato, and A. Sugimoto, "Social group discovery from surveillance videos: A data-driven approach with attention-based cues," in *British Machine Vision Conference*, 2013, pp. 1–11.

[51] J. Sochman and D. C. Hogg, "Who knows who - inverting the social force model for finding groups," in *IEEE International Conference on Computer Vision Workshops*, 2011, pp. 830–837.